(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
1 May 2003 (01.05.2003)

PCT

(10) International Publication Number
WO 03/036508 A2

(51) International Patent Classification[7]: G06F 15/78

(21) International Application Number: PCT/US02/33321

(22) International Filing Date: 18 October 2002 (18.10.2002)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/345,315 22 October 2001 (22.10.2001) US

(71) Applicant *(for all designated States except US)*: SUN MICROSYSTEMS, INC. [US/US]; 4150 Network Circle, Santa Clara, CA 95054 (US).

(72) Inventor: KOHN, Leslie, D.; 43967 Rosemere Drive, Fremont, CA 94539 (US).

(74) Agent: LEAVELL, George, B.; Martine & Penilla, LLP, Suite 170, 710 Lakeway Drive, Sunnyvale, CA 94085 (US).

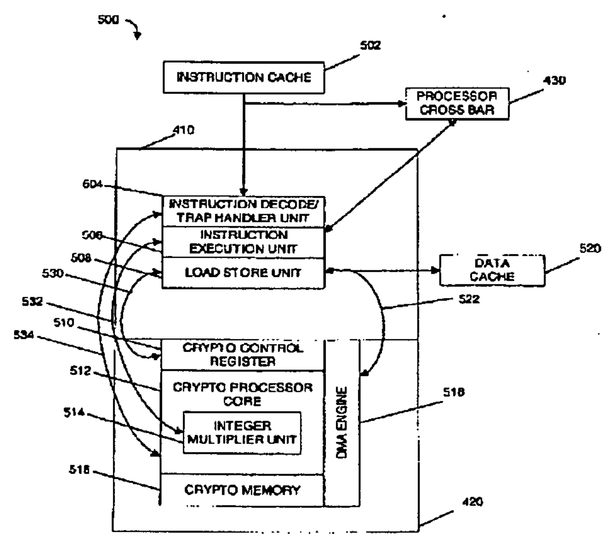
(81) Designated States *(national)*: DE, GB, JP, KR.

(84) Designated States *(regional)*: European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, SK, TR).

Published:
without international search report and to be republished upon receipt of that report

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: STREAM PROCESSOR WITH CRYPTOGRAPHIC CO-PROCESSOR



(57) Abstract: A microprocessor includes a first processing core, a first cryptographic co-processor and an integer multiplier unit that is coupled to the first processing core and the first cryptographic co-processor. The first processing core includes an instruction decode unit, an instruction execution unit, a load/store unit. The first cryptographic co-processor is located on a first die with the first processing core. The first cryptographic co-processor includes a cryptographic control register, a direct memory access engine that is coupled to the load/store unit in the first processing core and a cryptographic memory.

WO 03/036508 A2

STREAM PROCESSOR WITH CRYPTOGRAPHIC CO-PROCESSOR

BACKGROUND OF THE INVENTION

1. Field of the Invention

[1] The present invention relates generally to microprocessors, and more particularly, to microprocessors that include a cryptographic co-processor within the microprocessor die.

2. Description of the Related Art

[2] Server computers (i.e., servers) process all sorts of data transactions. One common type of data transaction is an encrypted data transaction that typically requires the server to perform at least one of an encryption function and a decryption function. Figure 1 shows a typical server 102 and client computer 110 that are linked by a network 104, such as the Internet or other network.

[3] Figure 2 is a high-level block diagram of a typical server 102. As shown, the server 102 includes a processor 202, ROM 204, and RAM 206, each connected by a peripheral bus system 208. The peripheral bus system 208 may include one or more buses connected to each other through various bridges, controllers and/or adapters, such as are well known in the art. For example, the peripheral bus system 208 may include a "system bus" that is connected through an adapter to one or more expansion buses, such as a Peripheral Component Interconnect (PCI) bus. Also coupled to the peripheral bus system 208 are a mass storage device 210, a network interface 212, a number (N) of input/output (I/O) devices 216-1 through 216-N and a peripheral cryptographic processor 220.

[4] I/O devices 216-1 through 216-N may include, for example, a keyboard, a pointing device, a display device and/or other conventional I/O devices. Mass storage device 210 may include any suitable device for storing large volumes of data, such as a magnetic disk or tape, magneto-optical (MO) storage device, or any of various types of Digital Versatile Disk (DVD) or Compact Disk (CD) based storage.

[5] The peripheral cryptographic processor 220 (i.e., crypto-processor) is linked to the processor 202 by the peripheral bus system 208. The crypto-processor 220 performs encryption and decryption operations that may be necessary for encrypted data transactions such as between the server 102 and the client 110. In some servers the crypto-processor 220 can also be external to the server 102 and linked to the processor 202 by one of the I/O devices 216-1 through 216-N.

[6] Network interface 212 provides data communication between the computer system and other computer systems on the network 104. Hence, network interface 212 may be any device suitable for or enabling the server 102 to communicate data with a remote processing system (e.g., client computer 110) over a data communication link, such as a conventional telephone modem, an Integrated Services Digital Network (ISDN) adapter, a Digital Subscriber Line (DSL) adapter, a cable modem, a satellite transceiver, an Ethernet adapter, or the like.

[7] Typically the processor 202 can operate at clock speeds of up to or more than 1 GHz. Conversely, the peripheral bus system 208 typically operates at a substantially slower speed such as about 166 MHz or similar. Further, the crypto-processor 220 typically operates at a speed similar to the peripheral bus system 208. This is because the crypto-processor 220 cannot process data any faster than the data can be transported across the peripheral bus system 208. Further, the crypto-processor 220 is typically a customized, specialized processor (i.e. an application specific integrated circuit (ASIC)) that may not be made by the latest, highest performance manufacturing technologies and therefore the maximum processing speed (i.e., the crypto-processor clock speed) of the crypto-processor 220 is substantially less than the maximum processing speed of the processor 202.

[8] Figure 3 is a flowchart diagram of the method operations 300 of a typical encrypted data transaction within the server 102. The encrypted data transaction can be any data transaction that required encryption, decryption or both encryption and decryption such as an e-commerce transaction between the server 102 and the client computer 110. In operation 305, data is received in the server 102 such as from the client computer 110 or because of a request by the client computer 110.

[9] In operation 310, the received data is analyzed to determine if the received data is encrypted. For example, the data may be encrypted because the data includes a user's personal and/or financial data or other data that is transported during an encrypted session such as SSL (secure sockets layer) or other encryption methods.

[10] If the received data is found to not be encrypted data, in operation 310, then the received data is processed as described in operation 330 below. Alternatively, if, in operation 310, the received data is determined to be encrypted data, then, in operation 315, the encrypted data is sent to the peripheral crypto processor 220 via the peripheral bus system 208.

[11] In operation 320, the crypto processor 220 decrypts the encrypted data. In operation 325, the crypto processor 220 outputs the decrypted data to the processor 202 via the peripheral bus system 208. In operation 330, the processor 202 processes the data to produce result data.

[12] In operation 335, the result data is analyzed to determine if the result data should be encrypted. If the result data does not require encryption, then the processor outputs the result data to the client 110, in operation 340, and the method operations end. Alternatively, if, in operation 335, the result data required encryption, then in operation 345, the processor outputs the result data to the crypto-processor via the peripheral bus system 208.

[13] In operation 350, the crypto processor 220 encrypts the result data. In operation 355, the crypto processor 220 outputs the encrypted result data to the processor 202 via the peripheral bus system 208. In operation 360, the processor outputs the encrypted result data to the client 110 and the method operations end.

[14] Transferring the data to be encrypted, decrypted or processed between the crypto processor 220 and the processor 202 is very slow. Further, the slower processing speed of the crypto processor 220 also limits the rate at which the data is encrypted or decrypted. Further, if a large volume of data such as streaming data (e.g., streaming audio, streaming video, etc.) is being encrypted and/or decrypted then the rate the server 102 can serve the streaming data is limited by the rate at which the streaming data can be encrypted and/or decrypted. Further still, the multiple transfers of the streaming data between the crypto

processor 220 and the processor 202 can dominate the usage of the peripheral bus system 208 and the I/O systems inside the crypto processor 220 and the processor 202, thereby limiting further the ability of the processor 202 to perform any functions other than transferring data to and from the crypto processor 220.

[15] In view of the foregoing, there is a need for a system and method for increased and/or more efficient data encryption and decryption process speeds.

SUMMARY OF THE INVENTION

[16] Broadly speaking, the present invention fills these needs by a system and method for increased and/or more efficient data encryption and decryption process speeds. It should be appreciated that the present invention can be implemented in numerous ways, including as a process, an apparatus, a system, computer readable media, or a device. Several inventive embodiments of the present invention are described below.

[17] One embodiment includes a microprocessor includes a first processing core, a first cryptographic co-processor and an integer multiplier unit that is coupled to the first processing core and the cryptographic co-processor. The first processing core includes an instruction decode unit, an instruction execution unit, a load/store unit. The first cryptographic co-processor is located on a first die with the first processing core. The cryptographic co-processor includes a cryptographic control register, a direct memory access engine that is coupled to the load/store unit in the first processing core and a cryptographic memory.

[18] The integer multiplier unit can be included within the first processing core or within the first cryptographic co-processor.

[19] The cryptographic memory is at least large enough to perform a Montgomery multiplication function.

[20] In one embodiment, the integer multiplier unit is a 64-bit X 64-bit multiplier unit.

[21] The load/store unit can be coupled to a main memory system hierarchy.

[22] The first processing core is coupled to a second processing core by a processor crossbar. The second processing core is coupled to a second cryptographic co-processor that

is located on a second die with the second processing core. Alternatively, the second processing core and the second cryptographic co-processor can be located on the first die.

[23] The first cryptographic co-processor can be coupled to the instruction decoder unit.

[24] The first cryptographic co-processor and the first processing core share the integer multiplier unit.

[25] The direct memory access engine can be coupled to the load/store unit by a 64-bit data bus.

[26] The cryptographic control register can include data that identifies a type of cryptographic instruction received in the first cryptographic co-processor.

[27] One alternative embodiment includes a method of executing a cryptographic command. A cryptographic instruction is received in an load store unit in a processing core on a first die. The cryptographic instruction is analyzed to determine if the cryptographic instruction is a crypto store instruction. If the cryptographic instruction is a crypto store instruction, then a source operand of the crypto store instruction is stored in a crypto control register in a cryptographic co-processor on the first die. The source operand is analyzed to determine if the source operand identifies a corresponding crypto command. If the source operand identifies the corresponding crypto command, the corresponding crypto command is executed.

[28] The cryptographic co-processor can also send an interrupt to an instruction execution unit that is included in the processing core such as when execution of a crypto command is completed.

[29] A result of the cryptographic instruction can also be output to a memory system using a load store unit that is included in the processing core.

[30] Execution of the cryptographic instruction in the cryptographic co-processor can also include accessing data through the load store unit.

[31] The cryptographic co-processor can also include a direct memory access engine. Accessing data can also include loading and storing data in a main memory.

[32] Executing the cryptographic instruction in the cryptographic co-processor can also include executing a multiplication function. The cryptographic co-processor can include an integer multiplier unit for executing the multiplication function.

[33] The various embodiments of the present invention provide the ability for a crypto processor to rapidly encrypt and/or decrypt data, such as streaming data, at rates much greater than possible by a prior art crypto processor.

[34] Other aspects and advantages of the invention will become apparent from the following detailed description, taken in conjunction with the accompanying drawings, illustrating by way of example the principles of the invention.

BRIEF DESCRIPTION OF THE DRAWINGS

[35] The present invention will be readily understood by the following detailed description in conjunction with the accompanying drawings, and like reference numerals designate like structural elements.

[36] Figure 1 shows a typical server and client computer that are linked by a network, such as the Internet or other network.

[37] Figure 2 is a high-level block diagram of a typical server.

[38] Figure 3 is a flowchart diagram of the method operations of a typical encrypted data transaction within the server.

[39] Figure 4 shows a single CPU die (chip) in accordance with one embodiment of the present invention.

[40] Figure 5 shows a detailed view of the processor core and cryptographic co-processor in accordance with one embodiment of the present invention.

[41] Figure 6 is a flowchart 600 of the method operations of the paired processor 410 and crypto co-processor 420 according to one embodiment of the present invention.

DETAILED DESCRIPTION OF THE EXEMPLARY EMBODIMENTS

[42] Several exemplary embodiments for a system and method for increased and/or more efficient data encryption and decryption process speeds will now be described. It will be apparent to those skilled in the art that the present invention may be practiced without some or all of the specific details set forth herein.

[43] Figure 4 shows a single CPU die (chip) 400 in accordance with one embodiment of the present invention. The CPU chip 400 includes a processing core 410. The processing core 410 is paired with a cryptographic co-processor 420. The cryptographic co-processor 420 is optimized to minimize the amount of hardware added to the CPU die 400. The CPU die 400 can also include additional processing cores 410n and each of the additional processing cores 410n is also paired with a cryptographic co-processor 420n. The processing cores 410, 410n are electrically coupled by a processor crossbar 430. The processor crossbar 430 is a data bus system that provides a common data communication link between the processing cores 410, 410n and other common devices that may be accessed by the processing cores 410, 410n such as memory systems and input/output (I/O) systems.

[44] Figure 5 shows a detailed view of the processor core 410 and cryptographic co-processor 420 in accordance with one embodiment of the present invention. The processor core 410 includes an instruction decode/trap handler unit 504, an instruction execution unit 506, a load store unit 508 and an integer multiplier unit 514. An instruction cache 502 is coupled to the input of the instruction decode/trap handler unit 504. A data cache 520 is coupled to the load store unit 508. The data cache 520 and the instruction cache 502 are also coupled to the processor crossbar 430. The data cache 520 can be a level-1 cache and can be between about 4kb and about 64kb in size.

[45] The crypto-coprocessor 420 includes a crypto control register 510, a crypto memory 516, a DMA engine 518 and the crypto co-processor core 512. The crypto control register 510 stores the settings of the crypto co-processor 420. The settings can include identifying the type of encryption or decryption command or operation to be performed. The types of encryption or decryption command can include any of the encryption and decryption schemes

known in the art. The crypto control register 510 also stores the status of the current crypto operations and is accessible by the processor 410 so that the processor 410 can check the status of the current crypto operations. The crypto control register 510 is linked to the load store unit 508 by a logical link 530, which represents the bi-directional interchange of data between the load store unit 508 and the crypto control register 510.

[46] The DMA engine 518 is coupled to the load store unit 508 by a crypto bus 522. The DMA engine 518 provides more direct access to the memory system hierarchy, such as the main memory, the data cache 520 and a level-2 cache, that can be accessed via the load store unit 508 and, if necessary, the processor crossbar 430. The crypto bus 522 can be as wide as feasible to enable rapid data transfer between the crypto coprocessor 420 and the processing core 410. In one embodiment, the crypto bus 522 is a 64-bit bus.

[47] The crypto memory 516 is sufficiently large enough to hold the operands and results for a particular crypto operation. By way of example, in an RSA decryption application, the crypto memory 516 is about 1.3 KB, which is large enough for a modular exponentiation on 2048-bit keys.

[48] As shown in Figure 5, the integer multiplier unit 514 is included within the crypto processor core 512 but directly accessible by the processor core 410 by a logical link 532. Alternatively, the integer multiplier unit 514 can be part of the processor core 410 as long as the crypto processor core 512 can directly access the integer multiplier unit 514. In this manner the crypto processor core 512 and the processor core 410 can share the integer multiplier unit 514 so as to reduce the space used (i.e., number of devices required) on the die. Typical, prior art crypto processors were not included on the die 400 with the processor core because the crypto processors consumed too much valuable space on the die that was needed more for the processor core.

[49] Sharing several components significantly reduces the "footprint" of the crypto processor so as to allow the crypto processor to be placed on the same die 400 as the processor core. The integer multiplier unit 514 performs the modular multiply and modular exponentiation functions. The processor core 410 only uses the integer multiplier unit 514 about 2-5% of the time. Therefore the crypto processor 420 can use the integer multiplier 514 about 95-98% of the time without impacting operations within the processor core 410.

Moving the integer multiplier unit 514 into the crypto processor 420 further streamlines the crypto functions of the integer multiplier unit 514.

[50] The integer multiplier unit 514 is capable of performing modular arithmetic such as Montgomery multiply functions and exponentiation. A Montgomery multiply function is a technique for performing modular multiplication on a large integer (e.g. a 2048 bit number) using two multiplications rather than a multiplication and a division. The integer multiplier unit 514 can be a 64-bit X 64-bit integer multiplier unit. A 64-bit X 64-bit integer multiplication unit can directly access operands that are stored in the crypto memory 516 rather than flooding the crypto bus 522 and the load store unit 508 every clock cycle. Flooding the crypto bus 522 and the load store unit 508 every clock cycle would effectively stall the processor core 410 because the load store unit 508 would only be able to address the demands of the crypto processor 420. Having a crypto memory 516 that is sufficiently large enough to perform a complete modular exponentiation relieves the data throughput load on the crypto bus 522 and the load store unit 508 and thereby allows the processor core 410 and the crypto processor 420 to operate in simultaneously on different operations and functions for many clock cycles.

[51] Figure 6 is a flowchart 600 of the method operations of the paired processor 410 and crypto co-processor 420 according to one embodiment of the present invention. The instruction cache 502 temporarily stores the next instruction to be executed in the processor core 410. In operation 605 the next instruction is received in the instruction decode/trap handler unit 504 from the instruction cache 502.

[52] The received instruction is forwarded to the instruction execution unit 506 for execution in operation 610. The instruction execution unit 506 analyzes the received instruction to determine if the received instruction is a load or a store instruction in operation 615.

[53] If, in operation 615 above, the received instruction is not a load or store instruction, the instruction is executed as required in the various stages 504, 506, 508 of the processor core 410 as necessary to complete execution of the non-load/non-store instruction, in operation 620 and the method operations on the executed instruction result ends.

[54] If, in operation 615 above, the received instruction is a load instruction or a store instruction, the received instruction is forwarded to the load store unit 508 in operation 630.

[55] In operation 635, the load store unit 508 analyzes the received instruction to determine if the received instruction is a crypto load instruction or a crypto store instruction.

[56] If, in operation 635, the received instruction is not a crypto load instruction or a crypto store instruction, the non-crypto store instruction/non-crypto load instruction is executed in the load store unit 508 (i.e., the prescribed load or store function is performed) in operation 640 and the method operations for that instruction ends.

[57] If, in operation 635, the received instruction is a crypto load instruction or a crypto store instruction, the crypto load or crypto store instruction is analyzed in operation 650.

[58] If, in operation 650, the crypto load or crypto store instruction is a crypto load instruction, the load/store unit 508 transfers the data from the crypto co-processor 420 back to the processor 410, in operation 655 and the method operations for the instruction end.

[59] If, in operation 650, the crypto load or crypto store instruction is a crypto store instruction (i.e., not a crypto load instruction), the load/store unit 508 transfers a source operand of the crypto store instruction to the crypto control register 510 in operation 660. In operation, 665, the crypto co-processor 420 analyzes the data in the crypto control registers 510 to determine if the stored data identifies a corresponding crypto command. If the data does not identify a corresponding crypto command, the method operations for the instruction end.

[60] If, in operation 665, the data identifies a corresponding crypto command, the corresponding crypto command is executed in operation 670.

[61] When the execution of the crypto command is complete, the crypto co-processor 420 can send an interrupt to the instruction execution unit 506 in the processor core 410 via the logical link 534, in operation 670 and the method operations for the crypto command ends. Some crypto command can also cause data in the crypto memory 516 to be output to the memory system via the load store unit 508. Conversely, some crypto commands will cause data in the memory system to be loaded into the crypto memory via the load store unit 508.

[62] The crypto co-processor 420 can process data independent of the processor core 410 for multiple and even thousands of clock cycles. Pairing the crypto processor 420 with the processing core 410 on the same die 400 increases the speed of the encryption and/or decryption processes by operating the crypto processor 420 and the processing core 410 at the same clock speed. The speed of the encryption and/or decryption processes is also increased because the data (e.g., data stream) to be encrypted and/or decrypted is not required to be transmitted the relatively long distance between the crypto processor 220 and the processor 202, at the much slower speed across the peripheral bus 208 such as described in Figure 2 above. Further, the pairing the crypto processor 420 with the processing core 410 allows the crypto processor 420 to directly access the memory system hierarchy through the crypto processor's 420 DMA engine 518 and the load store unit 508. This direct memory access allows the crypto processor 420 to rapidly encrypt and/or decrypt streaming data at rates much greater than possible by a peripheral crypto processor 220 shown in Figure 2 above.

[63] As used herein the term "about" means +/- 10%. By way of example, the phrase "about 250" indicates a range of between 225 and 275.

[64] With the above embodiments in mind, it should be understood that the invention might employ various computer-implemented operations involving data stored in computer systems. These operations are those requiring physical manipulation of physical quantities. Usually, though not necessarily, these quantities take the form of electrical or magnetic signals capable of being stored, transferred, combined, compared, and otherwise manipulated. Further, the manipulations performed are often referred to in terms, such as producing, identifying, determining, or comparing.

[65] It will be further appreciated that the instructions represented by the operations in Figure 6 are not required to be performed in the order illustrated, and that all the processing represented by the operations may not be necessary to practice the invention.

[66] Although the foregoing invention has been described in some detail for purposes of clarity of understanding, it will be apparent that certain changes and modifications may be practiced within the scope of the appended claims. Accordingly, the present embodiments are to be considered as illustrative and not restrictive, and the invention is not to be limited to

the details given herein, but may be modified within the scope and equivalents of the appended claims.

What is claimed is:

Claims

1. A microprocessor comprising:

a first processing core including:

an instruction decode unit;

an instruction execution unit;

a load/store unit;

a first cryptographic co-processor located on a first die with the first processing core; and

an integer multiplier unit that is coupled to the integer execution unit and the first cryptographic co-processor.

2. The microprocessor of claim 1, wherein the integer multiplier unit is included within the first processing core.

3. The microprocessor of claim 1, wherein the integer multiplier unit is included within the first cryptographic co-processor.

4. The microprocessor of claim 1, wherein the integer multiplier unit is a 64-bit X 64-bit multiplier unit.

5. The microprocessor of claim 1, wherein the load/store unit is coupled to a main memory hierarchy.

6. The microprocessor of claim 1, wherein the first processing core is coupled to a second processing core by a processor crossbar.

7. The microprocessor of claim 6, wherein the second processing core is coupled to a second cryptographic co-processor that is located on a second first die with the second processing core.

8. The microprocessor of claim 6, wherein the second processing core and the second cryptographic co-processor are located on the first die.

9. The microprocessor of claim 1, wherein the first cryptographic co-processor is coupled to the load store unit.

10. The microprocessor of claim 1, wherein the first cryptographic co-processor and the first processing core share the integer multiplier unit.

11. The microprocessor of claim 1, wherein the first cryptographic co-processor includes:

a cryptographic control register;

a direct memory access engine that is coupled to the load/store unit;

a cryptographic memory; and

12. The microprocessor of claim 11, wherein the cryptographic memory is at least large enough to perform a Montgomery multiplication function.

13. The microprocessor of claim 11, wherein the direct memory access engine is coupled to the load/store unit by a 64-bit data bus.

14. The microprocessor of claim 11, wherein the cryptographic control register includes data that identifies a type of cryptographic command received in the first cryptographic co-processor.

15. A method of executing a cryptographic command comprising:

receiving a cryptographic instruction in an load store unit in a processing core on a first die;

determining if the cryptographic instruction is a crypto store instruction;

if the cryptographic instruction is a crypto store instruction, then a source operand of the crypto store instruction is stored in a crypto control register in a cryptographic co-processor on the first die;

determining if the source operand identifies a corresponding crypto command; and

executing the corresponding crypto command if the source operand identifies the corresponding crypto command.

16. The method of claim 15, further comprising sending an interrupt from the cryptographic co-processor to an instruction execution unit that is included in the processing core.

17. The method of claim 15, further comprising outputting a result of the cryptographic instruction to a memory system using a load store unit that is included in the processing core.

18. The method of claim 15, wherein executing the cryptographic instruction in the cryptographic co-processor includes:

accessing data through the load store unit.

19. The method of claim 18, wherein the cryptographic co-processor includes a direct memory access engine.

20. The method of claim 19, wherein the cryptographic co-processor includes an integer multiplier unit.

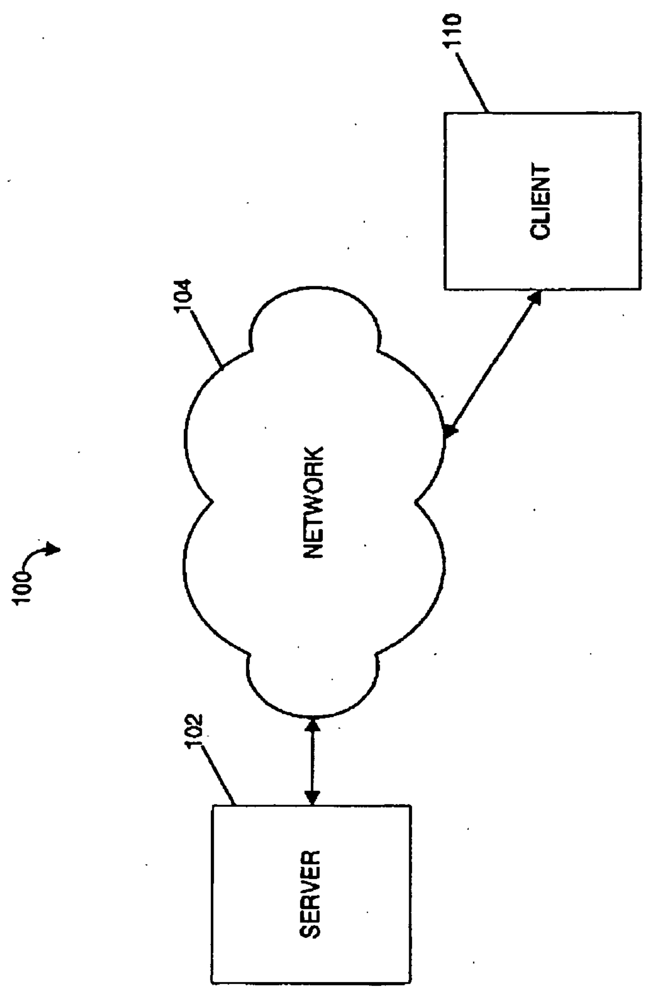


FIGURE 1
PRIOR ART

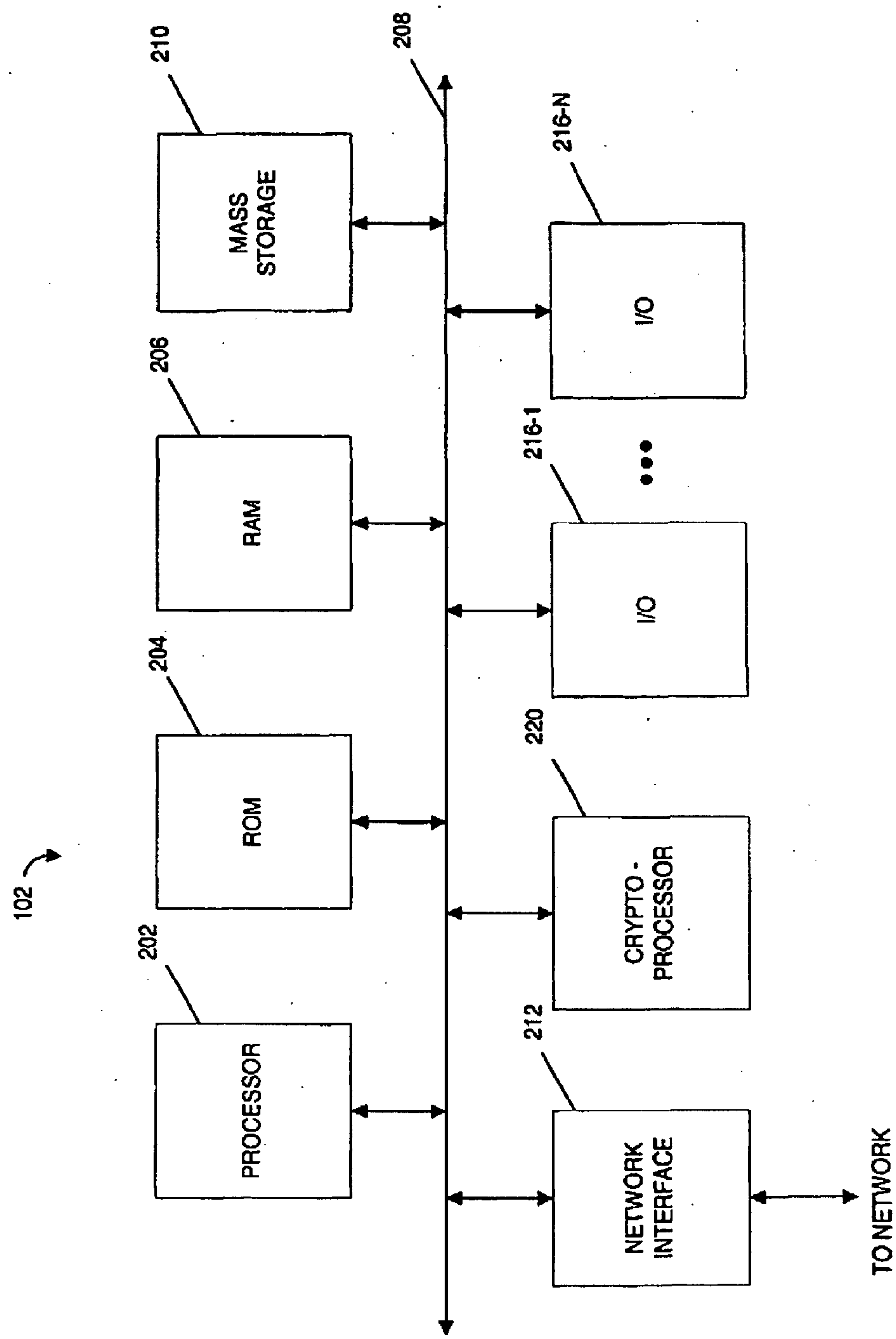


FIGURE 2
PRIOR ART

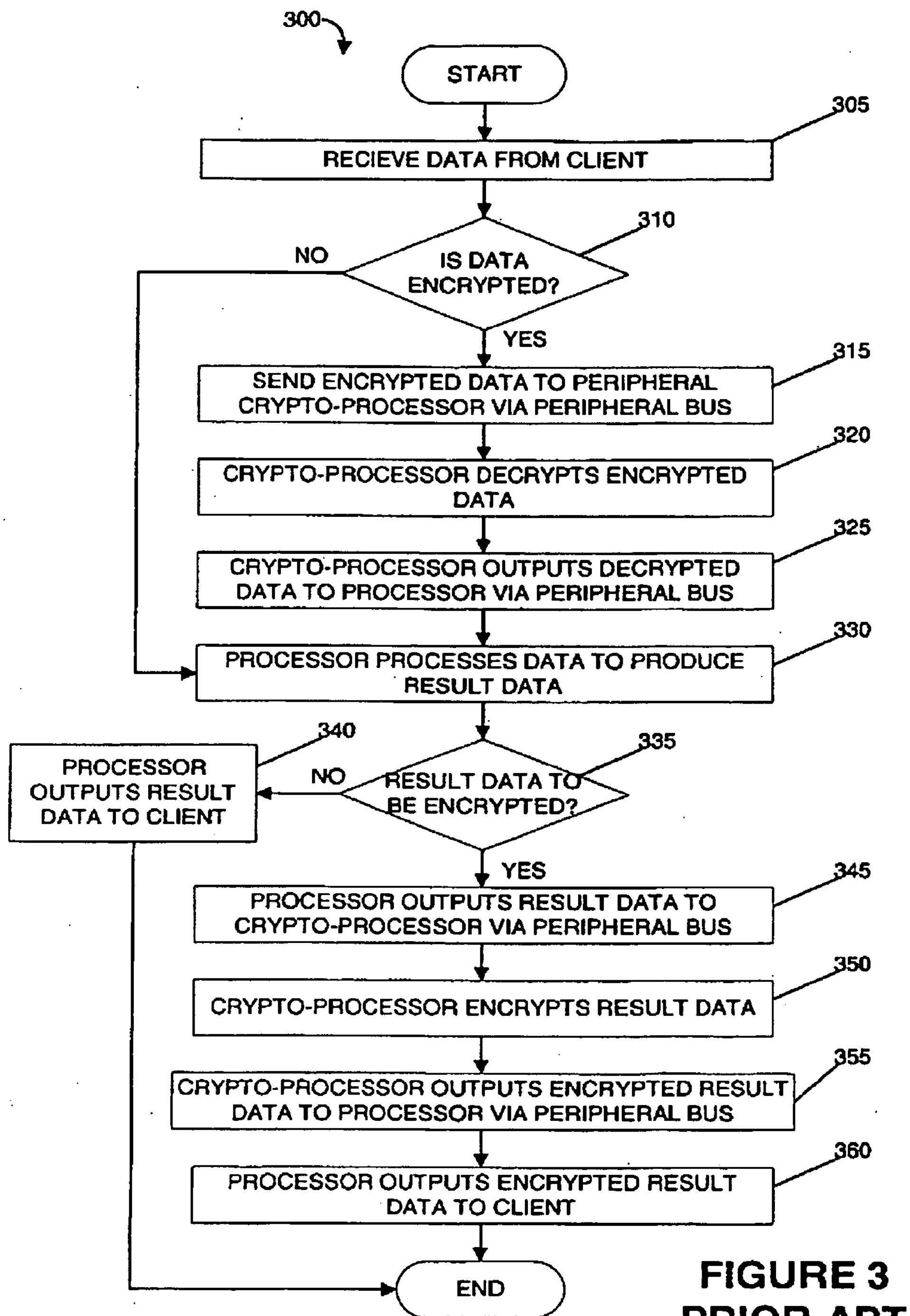


FIGURE 3
PRIOR ART

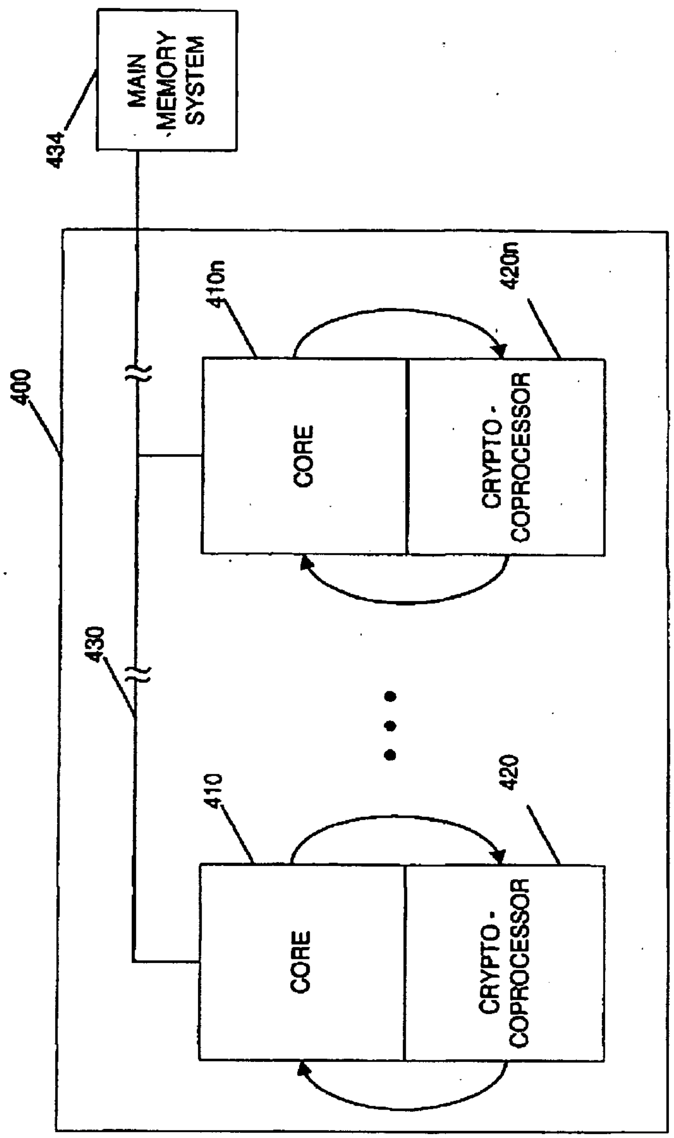

MAIN MEMORY SYSTEM
434
400
410n
420n
CORE
CRYPTO-COPROCESSOR
430
410
420
CORE
CRYPTO-COPROCESSOR

FIGURE 4

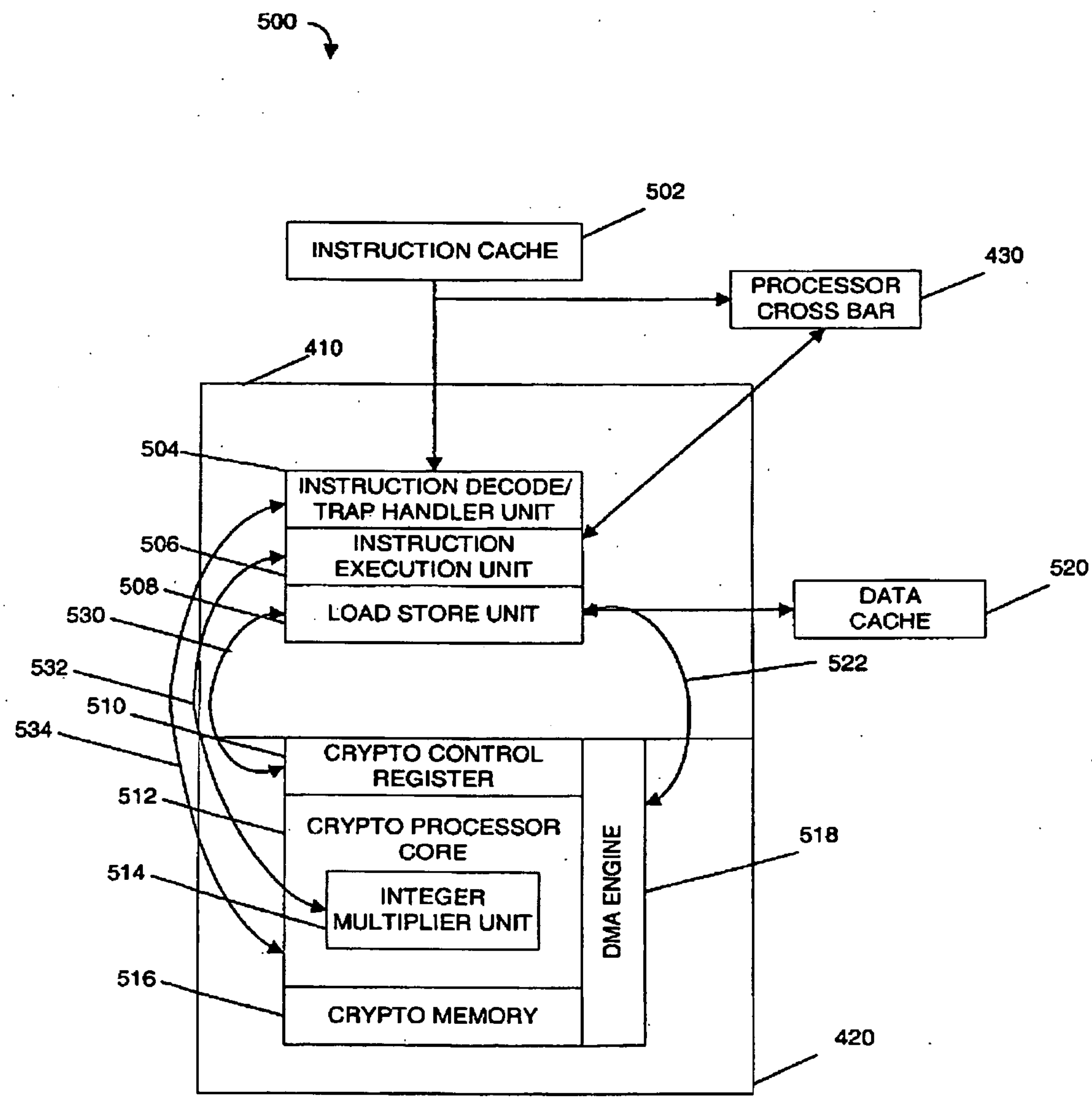
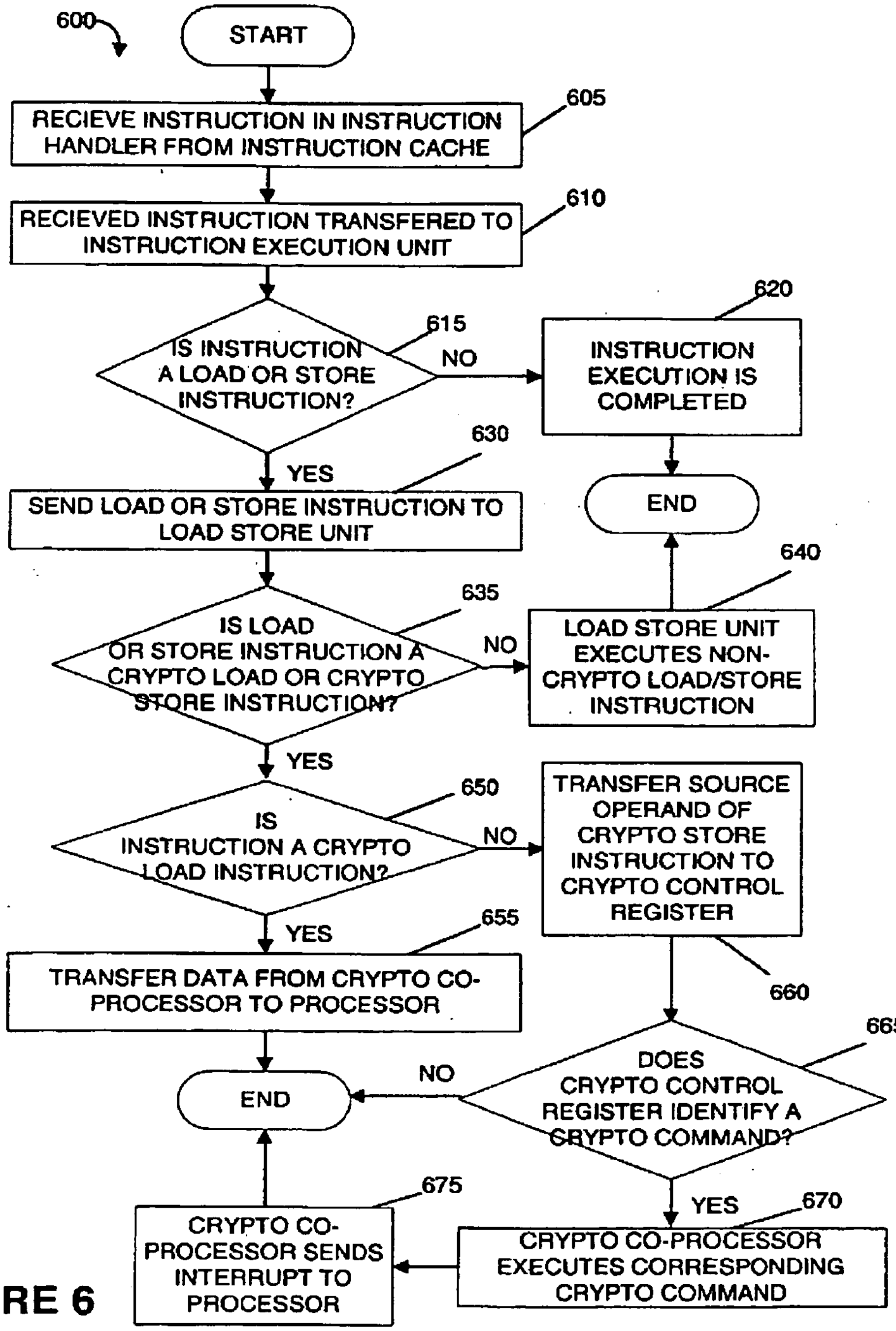

500
502
INSTRUCTION CACHE
430
PROCESSOR CROSS BAR
410
504
INSTRUCTION DECODE/ TRAP HANDLER UNIT
506
INSTRUCTION EXECUTION UNIT
508
LOAD STORE UNIT
530
520
DATA CACHE
522
532
510
534
CRYPTO CONTROL REGISTER
512
CRYPTO PROCESSOR CORE
514
INTEGER MULTIPLIER UNIT
516
CRYPTO MEMORY
DMA ENGINE
518
420

FIGURE 5



600
START
RECIEVE INSTRUCTION IN INSTRUCTION HANDLER FROM INSTRUCTION CACHE
605
RECIEVED INSTRUCTION TRANSFERED TO INSTRUCTION EXECUTION UNIT
610
IS INSTRUCTION A LOAD OR STORE INSTRUCTION?
615
NO
INSTRUCTION EXECUTION IS COMPLETED
620
END
YES
SEND LOAD OR STORE INSTRUCTION TO LOAD STORE UNIT
630
IS LOAD OR STORE INSTRUCTION A CRYPTO LOAD OR CRYPTO STORE INSTRUCTION?
635
NO
LOAD STORE UNIT EXECUTES NON-CRYPTO LOAD/STORE INSTRUCTION
640
YES
IS INSTRUCTION A CRYPTO LOAD INSTRUCTION?
650
NO
TRANSFER SOURCE OPERAND OF CRYPTO STORE INSTRUCTION TO CRYPTO CONTROL REGISTER
660
YES
TRANSFER DATA FROM CRYPTO CO-PROCESSOR TO PROCESSOR
655
END
NO
DOES CRYPTO CONTROL REGISTER IDENTIFY A CRYPTO COMMAND?
665
YES
CRYPTO CO-PROCESSOR EXECUTES CORRESPONDING CRYPTO COMMAND
670
CRYPTO CO-PROCESSOR SENDS INTERRUPT TO PROCESSOR
675

FIGURE 6